DOCUMENT RESUME

ED 138 290 IR 004 719

AUTHOR Pelczarski, Mark; And Others
TITLE INFOSH--A Basic Program for File Search. Illinois Series on Educational Application of Computers. Number 13.
PUB DATE Nov 76
NOTE 21p.; For related documents see IR 004 715-716 ; Appendices may be marginally legible due to print quality of the original

EDRS PRICE MF-$0.83 HC-$1.67 Plus Postage.
DESCRIPTORS *Computer Programs; *Data Processing; *Information Retrieval; Input Output; Programing
IDENTIFIERS *BASIC; INFO; INFOSH

ABSTRACT

INFOSH is a program which allows the user to search through data in any file set up through the main INFO programs (IR 004 715) and obtain results of each search in a table format. It functions primarily to facilitate answering questions about data in a file in a way that does not involve manual scanning. It can give answers to questions dealing with trying to identify specific file items that satisfy one or more different conditions. This report describes what INFOSH can and can not do and how to use it. Appendices provide a list of variables used in INFOSH and their functions, a flow chart in INFOSH operation, and sample runs and listings. (DAG)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED138290

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Number 13

The Illinois Series on Educational Applications of Computers

INFOSH - A BASIC Program for File Search

PERMISSION TO REPRODUCE THIS COPYRIGHTED MATERIAL HAS BEEN GRANTED BY
Bruce Hicks
TO ERIC AND ORGANIZATIONS OPERATING UNDER AGREEMENTS WITH THE NATIONAL INSTITUTE OF EDUCATION. FURTHER REPRODUCTION OUTSIDE THE ERIC SYSTEM REQUIRES PERMISSION OF THE COPYRIGHT OWNER

Viktors J. Muiznieks, Clara Symonds*
and Mark W. Pelczarski

November 1976

College of Education
Department of Secondary Education
University of Illinois
Urbana, Illinois

BEST COPY AVAILABLE

© Bruce Hicks (1976)

*Maine Township West High School, Des Plaines, Illinois

IR004719

Preface

We examined some of the different kinds of searches which various users might want to perform on data stored in files produced by INFO (see ISEAC papers 9 and 10), and implemented the most important search types in the present version of the INFOSH program. The discussion in this paper reflects some of the ideas and concepts upon which the structure and formation of the INFOSH program are based.

Some familiarity with the INFO, INFOAD, and INFOPR programs previously discussed in ISEAC papers 9 and 10 would be helpful in understanding parts of this discussion, even though such knowledge is not essential for being able to capture the flavor of what the INFOSH program is and does.

Number 13

Table of Contents

1. What is INFOSH--INFO Search?..... p. 1
2. Why might someone need to use INFOSH?..... p. 1
3. What can INFOSH do?..... p. 2
4. What can't INFOSH do?..... p. 3
5. What do you need to do before you can use INFOSH?..... p. 4
6. How do you use INFOSH?..... p. 4

Appendices

A. List of variables used in INFOSH and their uses..... A1
B. Uncommon BASIC functions and statements used in INFOSH..... B1
C. Flow chart of INFOSH operation..... C1
D. Sample runs using INFOSH..... D1-D4
E. Listing of INFOSH program..... E1-E2

1. What is INFOSH--INFO Search?

The INFOSH program is an addition to the INFO package described in ISEAC papers 9 and 10. The program allows the user to search through the data in any file set up through the main INFO programs (INFO and INFOAD), and obtain the results of each search in a table format similar to that produced by the INFOPR program. The main function of the program is to facilitate answering questions about the data stored in the file in a way which does not involve manually scanning the listing produced by INFOPR.

Inspection by hand of the output from this search program is, of course, required, to actually answer many questions about the data because each individual user will ask many questions that are needed solely for their own individual purposes. Also, in general, no computer program can be as useful as the "eye-balling" capability of a human being who is scanning the data and at the same time making observations and judgments about the data. We therefore emphasize that a search program facilitates the search and analysis of the data-base by doing the part of the job the computer does best

2. Why might someone need to use INFOSH?

As intimated in the previous section, it appears to be the case that after a data file has been created through the INFO package, it becomes desirable to be able to do more with that file than simply store, update, and list the data in the file. Questions concerning the characteristics of the data arise, and so some means for answering those questions is required. INFOSH is a step in this direction. The INFOSH program examines the data stored in a file, and based upon the specifications set down by the user, produces a listing of all of the attributes the user desires for all of the items that satisfy the stated conditions. Without INFOSH, the only way to answer questions dealing with the regularities of the data stored in the file is by manually scanning the output from the INFOPR program, and performing careful comparisons by hand; this clearly becomes more and more untenable as the size of the data file increases.

The types of questions for which INFOSH is designed are questions dealing with trying to identify the specific items in the file that satisfy one or more different conditions; for example, INFOSH could be used to find the names of all the female students who have at least a B average in a student record file that has data on sex, grades, and student names, or it could be used to find all of the items in a parts catalog file that are produced by a certain manufacturer, if that data file contains information on manufacturer names and part types.

Another useful application of the INFOSH program, when used in conjunction with the rest of the programs in the INFO package, is the ability to simulate the use of a data-base with a data management system, so that questions pertaining to the particular data-base can be examined, and problem areas identified, before extensive design procedures and programming tasks are actually undertaken. Questions such as "What kinds of data should be included?," "How should the data be represented?," and "What kinds of questions are users of the data-base likely to ask about the data?" (so, "What kinds of search capabilities will be required of the system?") can be researched. Small test files containing representative headings and data can be easily entered, tested, and modified, and various types of searches can be performed on the data to see if relevant information can be retrieved without much difficulty, or if search options not available in INFOSH are required. By analyzing which aspects of the INFO package are particularly

advantageous, which ones are not required, and which ones are missing that will be needed, a more effective data-management system can be designed which meets the requirements of the particular data-base and its users. The management system can be designed in a faster and more direct fashion as a result of the experiences gathered while using the simulation model; also, the structure, form, and content of the data itself can be adjusted to best fit the needs of the system.

3. What Can INFOSH do?

Basically, INFOSH can be used to give answers to two types of questions about the data stored in a file. These questions take either the AND form--"Give me information on items satisfying (this and this and that and that) criteria," or the OR form--"Give me information on items satisfying (this or this or that or that) criteria." Questions which can be stated in one of these forms, within the guidelines set forth by the INFOSH program (i.e., the different search types), are the ones which can be answered by the INFOSH program.

There are five different kinds of searches that INFOSH can perform. The first four of these require the user to specify a standard upon which to base search comparisons; the standard is a string of letters and/or numbers which could be a word, a single letter or number, or a carefully chosen sequence of letters and/or numbers which are particularly useful in separating different types of items within the file, against which the data items in the file are to be judged. Searches are conducted under one heading at a time, and different types of searches for different (or even the same) heading are allowed. A search is not complete until all of the required criteria specified by the user are met. If the user decides that when any of the specified criteria is met the data item qualifies for inclusion in the output set, then as soon as any criteria under any heading is met, the data item is included (an ANY search is used to answer questions of the OR type); if the user decides that all of the criteria must be met before an item is included, then as soon as the criteria under one heading are satisfied by a data item, the rest of the headings for that item are checked to see if the data under each heading also satisfied the specified requirements, and if all the requirements are met, the data item is included (an ALL search is used to answer questions of the AND type).

The different kinds of searches which can be conducted under a heading are the following:

1) See if the data item matches a standard exactly.

2) See if the first part of the data item is exactly the same as a standard.

3) See if the data item is alphabetically or numerically less than or equal to a specified standard.

4) See if the data item is alphabetically or numerically greater than or equal to a specified standard.

5) See if the data item has a numerical equivalent that is a positive integer or zero.

The standard used in the first four search types is used as a basis for making judgments about certain data items. If, for example, you wanted to find all the data items in the list:

"ABEL" "ABRAHAM" "LINCOLN" "WASHINGTON" "WEBSTER"

that started with the letters "AB", you could choose a type 2 search and specify AB as the standard upon which to base the search. The search would then retrieve the items "ABEL" and "ABRAHAM" as satisfying the requirements.

The type 5 search does not require a standard to be specified; asking for this type of search also produces an added benefit--the total for all the numbers found under the heading that meet the rest of the search specifications is printed at the end of the output listing (for a more complete discussion of this search type, see section 6).

Before starting the search, the user has the option of specifying which headings for the data items found that satisfy the search conditions should be printed. These do not have to be the same as the headings searched under (see Appendix D for examples). For instance, a search might specify that the data items included should have a type 1 match with the standard FEMALE under the sex heading, and that the data items should also have a type 4 match with the standard C under the grade heading, and that the printout should include the information stored in each data item under the name and grade headings only. This search would produce a listing of the names of all the female students whose grades were C or lower.

Regardless of the type of search conducted, the program will always inform the user of how many items met the required search specifications, and the total number of items there were in the file examined.

The almost unlimited number of combinations and permutations of the above search types, their associated standards, and the ALL and ANY search satisfaction requirements, coupled with the fact that a search may look for two or more criteria within the same heading, identify the INFOSH program as a very powerful data analysis tool.

4. What can't INFOSH do?

INFOSH cannot answer all of the questions about the information stored in the file, but it can answer a lot of them. Questions which it cannot deal with are questions of the form "Give me information on items satisfying (this and this) or (that and that) criteria," or questions that take on even more complicated forms than this. Only questions of the simple AND and OR types mentioned in section 3 can be answered; no mixing of the AND and OR question types is allowed within a search. (This feature could be added to the program.)

Currently, to get answers to the more complex questions which involve combining the AND and OR question types, you have to make two or more separate searches (each asking only the "and" or "or" parts of the question), and then compare the results of those searches by hand (which, for large data-bases, is still easier than having to try and obtain the answers by looking at only a listing of the data as it appears in the INFOPR listing).

The hand-searching equivalent to the OR question type is the merging of all the items on the lists produced, while the AND equivalent is the identification of items common to all the produced lists.

Once again, it is worth emphasizing the point made in section 1, that INFOSH is primarily intended to be used as a tool for restricting the number of items in a file meeting certain specified criteria, and it is not designed to perform any of the evaluative or interpretive functions which a human researcher could perform once the pertinent data is made available.

There is a current restriction that a maximum of 50 search criteria can be specified for any one search; this number can be increased by slight modifications to the program (amounting to replacing all the 50's present in the program with the larger number of search criteria you want to be able to use).

5. What do you need to do before you can use INFOSH?

Almost nothing!! That is one of the programs greatest advantages; as soon as you have the INFO programs INFO, INFOAD, and INFOPR working, all of the files that are needed by INFOSH are already there. There is nothing special that you have to do each time before you can execute the INFOSH program.

Initially, of course, it may take some modifications to adopt the INFOSH program to your particular system, but this will probably involve incorporating the same kinds of changes it took to get the rest of the INFO package working, and retyping the statements that have a slightly different form or syntax in your version of the BASIC language. Appendix B tries to point out the most likely types of functions and statements that might have to be rewritten to conform to your requirements.

6. How do you use INFOSH?

To use INFOSH, you must first have a specific search question in mind that requires either the "anding" or "oring" of certain conditions contained under the headings in the file you wish to examine.

Some sample runs of INFOSH are given in Appendix D. These examples have been chosen to illustrate the description of the use of INFOSH given in this section, and should be looked at to clarify some of the topics briefly mentioned here. The examples are also useful for pointing out some of the implications and ramifications of the various search types.

What the INFOSH program will do is first ask you which file you want to search, and perform any password checks that are necessary. Then you are asked whether or not you need help. If you say you do need help, then a summary of the material presented in this paper is printed out (see Appendix E). After that, the program asks you if you want a list of the headings within the file you chose; if you respond YES, you get a listing of all the headings along with their corresponding heading numbers (#'s)*. The heading #'s become important later on, as we shall soon see.

The program then asks you if you want to see the search type #'s which correspond to the five search types discussed in section 3. If you respond YES, you get a listing of the different search types and their corresponding #'s (which will also be required later).

*Heading and search type numbers are designated by "#" both in this paper and in the INFOSH program itself.

Now you are ready to begin actually specifying your search. The program will ask you for the heading # you want to specify a search type for, and you will respond by entering in a heading #. The program will then ask for the search type # corresponding to the type of search you want done on this heading. If the type of search you specify requires a standard to be entered, the standard will also be asked for. After you enter in the standard, the procedure is repeated until you have entered in all the search conditions you want to specify. You indicate this to the program by typing in a heading # of 0(zero) when that question is asked.

If you asked for more than one condition in your search specification, then the program asks you if you want all or any of the search specification criteria to be met before a data item satisfying the search conditions is found. Specifying a 1, or that all conditions are to be met, corresponds to asking an AND question type, i.e., "Give me information about all the items satisfying condition 1 and condition 2 and condition 3, etc." Specifying a 2, or that any condition suffices to include a data item, corresponds to the OR question type, i.e., "Give me information about all the items satisfying condition 1 or condition 2 or condition 3, etc."

Finally, the program asks for the heading #'s of all the headings you want listed on the output for data items that are found to satisfy the search criteria. You specify one heading # at a time, and when you have finished specifying all the headings you want to include, you enter a heading # of 0(zero). If you want to include all of the headings, you can simply enter a -1 and all of the headings will be automatically included. Notice that performing a search for which all of the data items are found, and listing them all under each heading, produces the same results as asking for a listing from the INFOPR program.

Just before the data items satisfying the search conditions are listed, the program asks you for your name and the date, as well as the width of the paper on which the output is to be printed (in terms of the number of characters per line, with a maximum of 132 and a minimum of 1), and gives you a chance to roll the paper to the top of the next page so that the output can be neatly torn off and saved.

The tabular form for the output can be one of two kinds. The most common is the form in which the data headings serve as the column headings, just as in INFOPR. However, the headings can also be listed as they are in INFO when listing a particular data item, so that each line that is printed contains the heading name, and the corresponding data item; this second type of listing is the kind you get if you respond NO to the question asking about the column form listing of the output.

After the headings and data are listed on the terminal, the program prints out the number of items that satisfied the search criteria, as well as the total number of items in the file. If any of the search type #'s was a 5, the numerical total for the data items satisfying the search criteria under that heading is also printed.

The user can then specify if another search on the same file, or on another file is to be performed, or if some other program in the INFO package is to be executed, or if that is all for the time being, and that an exit from the program should take place.

The type 5 search has some peculiarities associated with it. First of all, no standard is required since the implicit standard is that you are looking for a number. In order for a number to meet the requirements of this search type, it must be a positive integer or 0(zero), i.e., a data item like "-5" would not be considered to satisfy the requirements of this search type, nor would an item like "12.3" be acceptable. Only "pure" integer numbers like "123" or "76" are acceptable, or numbers stored as "12 34", where there are blank spaces between the digits. For numbers of this latter type, the blanks are ignored, so that the above string would be converted into the number 1234, and added into the total as 1234.

These limitations are, of course, more restrictive than they need to be, and the program could be changed to recognize both negative numbers as well as numbers that contain a decimal point. The decision for not including these features was based on the assumption that the additional benefits that would result from being able to identify such numbers would not be enough to offset the disadvantages resulting from increasing the program's length, complexity, and transportability restrictions. (The various ways in which character strings can be manipulated on different implementations of the BASIC language, and the different conversion restrictions between strings, numbers, and/or ASCII characters that also exist, are what influenced the transportability related questions behind this decision.)

As a result of the above decisions, certain "tricks" can be used in storing data which can affect later searches in various ways. For example, if a file is created to keep track of information related to a particular set of items, say a parts listing for a certain kind of machine, then it may be valuable to include a heading of related costs. Under this heading all of the initial costs may be entered as integers (to the nearest dollar), while repair bills on these parts may be entered with the letter R in front of them. In this way, a search could be performed that would give the total initial cost of the parts by performing a type 5 search under the related costs heading; the total would not reflect any repair costs. Later, another search could be performed to get a listing of all the parts that needed repair by performing a type 2 search on the same heading, and specifying R as the standard.

Appendix A

Variable	Function(s) that variable or array serves
C	Specifies conditions of search; 1 all conditions required, 2 means any condition being satisfied will suffice.
H(50)	Contains heading #'s to be searched in user specified order.
I,J,S2,V1,V5,V6	Loop parameters.
I1	Number of headings to be searched for the search being conducted.
N2	Temporary counter for TABbing when printing table output.
N3	# of characters per column in table to be printed.
N4	Flag; 1-headings printed for table, 0-headings not yet printed.
N5	# of headings to be printed in the table.
N6	# of entries found meeting the search criteria.
N7	Flag; 1-headings by columns, 0-headings by rows (on printing).
N8	# of entries (total) in file.
N9	Used to check for null string data items.
P	Numerical equivalent of maximum length of data items in file.
R(50)	Type of search corresponding to each heading specified.
S1	# of headings in the file.
V2	Temporary counter for TABbing in printing table output.
V4	Flag; 1-at least one column total requested, 0-otherwise.
V7	Flag; 1-name, date, etc., printed already, 0-otherwise.
V(50)	Headings to be listed in the output printout; 1-print, 0-don't.
W(50)	Totals for each heading that a type 5 search was asked for.
A$	Temporary variable; also holds file description specification.
B$(50)	Standards for corresponding search types.
D$,Q$,X$,Y$,Z$	Input variables.
E$(50)	Contains the item information for each data entry currently being examined in the search.
F$	File name for file being searched.
N$	String equivalent of numeric variable P.
P$	Password for file being searched.
R$	Pattern string containing digits 0-9 and a blank.
S$(50)	Heading name for each heading.

Appendix B

Functions used that may not be standard in other BASIC systems:*

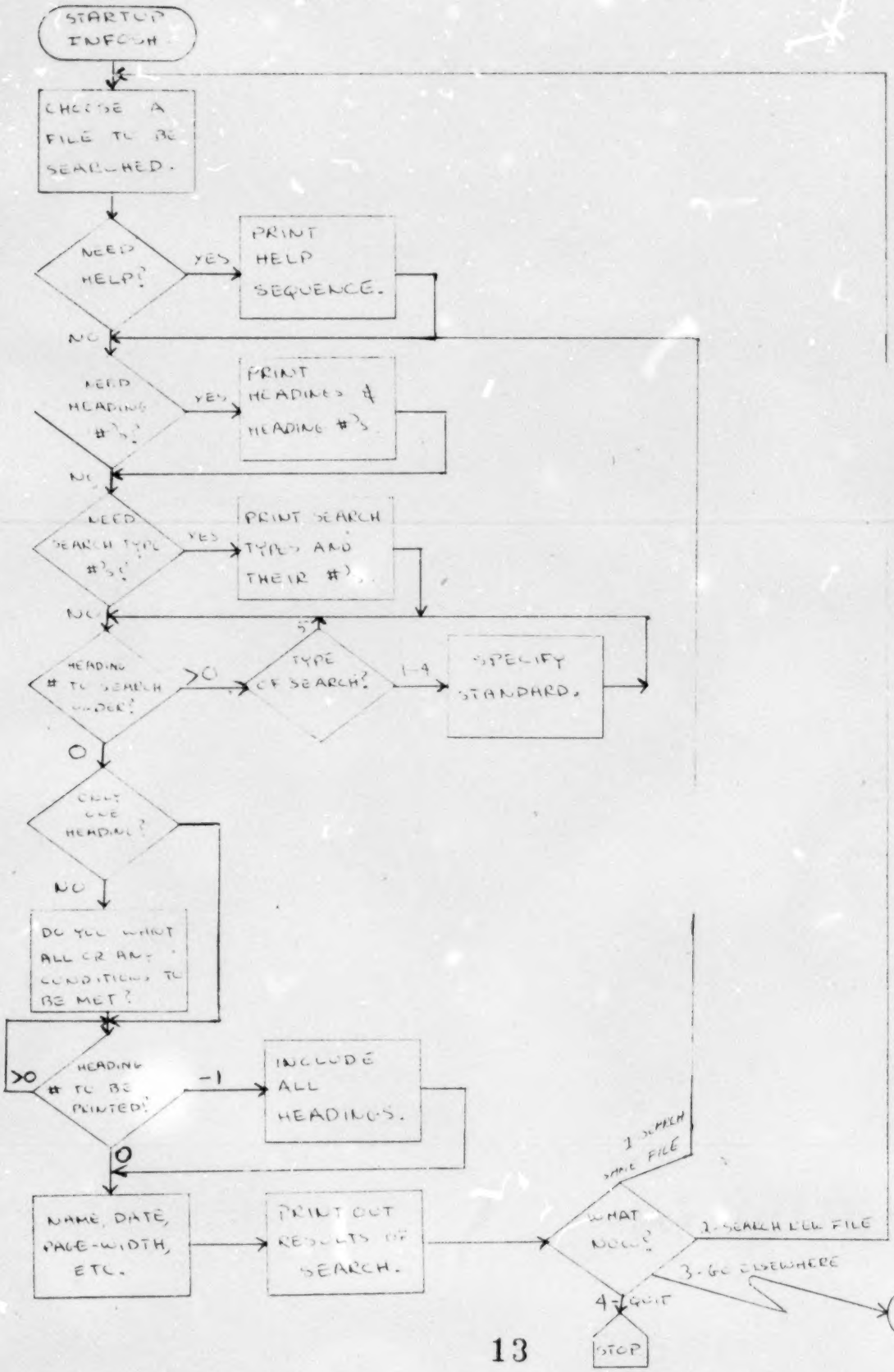
LEFT$(A$,L)	Returns the leftmost L characters of the string A$. E.G. LEFT$("ABCD",2) returns the string "AB".
MID$(A$,P,L)	Returns a string of L characters starting at position P from A$; if L is not specified, the rest of A$ starting at position P is returned. E.G. MID$("ABCD",2,2) returns string "BC", and MID$("ABCD",2) returns string "BCD".
VAL(A$)	Returns the numeric equivalent of string A$. E.G. VAL("123") returns the integer 123.

Statements used that may not be standard in other BASIC systems:

MARGIN W	Sets width of output to W characters per line. E.G. MARGIN 100 means each output line can be up to 100 characters long.
QUOTE #N	Convert all the items in the file on channel N to strings. E.G. QUOTE #3 would make any numbers in the file on channel 3 appear as their string equivalents, so 34, for example, would become "34".

*The INFOSH program was written using the same DEC-10 BASIC language as described in ISSAC paper #9 on INFO, and all of the limitations and restrictions described there apply to INFOSH as well.

APPENDIX C: BASIC FLOW OF INFOSH



NOTE: NOT PLACE INPUTS ARE ASKED FOR, THEY ARE CHECKED, IF POSSIBLE, FOR VALIDITY; IF AN UNACCEPTABLE RESPONSE IS FOUND, THE QUESTION (AND POSSIBLY AN ERROR MESSAGE) IS REPEATED UNTIL AN ACCEPTABLE ANSWER IS ENTERED. HENCE, SOME OF THE DECISION BOXES ONLY SHOW THE "VALID RESPONSE" ACTION, TAKEN, SINCE "INVALID RESPONSES" ARE NOT ALLOWED.

APPENDIX D

D1

SAMPLE RUNS

```
OLD INFOSH

READY
RUN

INFOSH          17:07          10-NOV-76


WHICH FILE DO YOU WANT? (TYPE 'NONE' FOR NONE)
CHOICES:
PARTS           CLASS
 ?PARTS

DO YOU NEED HELP(YES OR NO) ?YES

YOU WILL BE ASKED TO SPECIFY THE HEADING #'S OF ALL THE HEADINGS
YOU WANT TO SEARCH UNDER, AND THE PARTICULAR TYPE OF SEARCH YOU
WANT TO CONDUCT.  THE SEARCHES CAN BE OF THE TYPE (1) LOOK FOR ALL
EXACT MATCHES TO A SPECIFIED STANDARD UNDER THAT HEADING
(2) LOOK FOR ALL ITEMS UNDER THE HEADING THAT HAVE THE SAME FIRST
LETTERS AS A SPECIFIED STANDARD  (3) LOOK FOR ALL ITEMS UNDER THE
HEADING THAT HAVE A VALUE LESS THAN OR EQUAL TO A SPECIFIED STANDARD
(4) LOOK FOR ALL ITEMS UNDER THE HEADING THAT HAVE A VALUE GREATER
THAN OR EQUAL TO A SPECIFIED STANDARD  (5) LOOK FOR ALL ITEMS UNDER
THE HEADING THAT HAVE A NUMERICAL EQUIVALENT VALUE; BE CAREFUL
IF AN ENTRY CONTAINS 2 NUMBERS WITH BLANKS INBETWEEN, THEY WILL BE
TREATED AS ONE NUMBER--E.G.  '12  34' WOULD BE INTERPRETED AS 1234
AS A SIDE BONUS, YOU GET THE TOTAL FOR ALL OF THE NUMBERS FOUND
UNDER THIS HEADING THAT MEET THE REST OF THE SEARCH CRITERIA (IF ANY).
  ONCE YOU HAVE SPECIFIED ALL OF THE HEADINGS, SEARCH TYPES, AND
RELATED STANDARDS, YOU MUST ALSO INDICATE (1) IF ALL OF THE
CONDITIONS SPECIFIED MUST BE MET BEFORE ANY ITEM IS INCLUDED, OR IF
(2) MEETING ANY OF THE CRITERIA IS ENOUGH TO INCLUDE AN ITEM
(UNLESS YOU ARE ONLY SEARCHING UNDER ONE HEADING).  ALSO, YOU ARE
GIVEN THE CHOICE AS TO WHICH HEADINGS TO HAVE DISPLAYED FOR THE
ITEMS THAT MATCH THE ABOVE SPECIFICATIONS (THEY NEED NOT BE THE SAME
AS THE HEADINGS SEARCHED UNDER).  THE PROGRAM WILL ALSO ASK YOU IF
YOU WANT THE RESULTS LISTED IN TABULAR FORM (AS OPPOSED TO LINE FORM)
AND THE NUMBER OF CHARACTERS PER LINE THAT CAN BE PRINTED (1-132)
AS WELL AS YOUR NAME AND THE DATE.  IT ALLOWS YOU TO POSITION THE
PAPER TO THE TOP OF A NEW PAGE OF PAPER BEFORE PRINTING THE RESULTS OF
THE SEARCH, AND THEN GIVES YOU THE OPTIONS OF (1) SEARCHING THE SAME
FILE AGAIN  (2) SEARCHING ANOTHER FILE  (3) RETURNING TO THE
OTHER INFO PROGRAM OPTIONS  (4)  LEAVE THE PROGRAM AND RETURN TO
THE BASIC MONITOR.


DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?WHAT
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?Y

HERE ARE THE HEADINGS (AND THEIR CORRESPONDING NUMBERS)
 1              ITEM
 2              RELATED COSTS-#'S
 3              MANUFACTURER
 4              REPAIR SERVICE
 5              DESCRIPTION

DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?Y


CHOICES: 1) EXACT MATCH  2) BEGINNING MATCHES STANDARD
3) <= STANDARD  4) >= STANDARD  5) LOOK FOR NONNEGATIVE INTEGERS


NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?1
SEARCH TYPE # ?4
STANDARD ?AAAA

HEADING # ?0


NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
 ?-1
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?Y
WHAT IS YOUR NAME ?VIK MUIZNIEKS
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?NOVEMBER 10 1976
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?120
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?

FILE--PARTS   LISTED BY VIK MUIZNIEKS ON NOVEMBER 10 1976

ITEM                    RELATED COSTS-#'S     MANUFACTURER          REPAIR SERVICE        DESCRIPTION

CARD READER             3000                  DOCUMATION                                  OM 300L
LINE PRINTER            5000                  IBM                                         300 LINES/MINUTE
LINE PRINTER REP.       R100                                        IBM                 REPLACE PARTS
LINE PRINTER REP. 2     R50                                         IBM                 BROKEN PRINT HEADS
TTY1                    1000                  TELETYPE                                    MODEL ASR-33
TTY2                    1500                  DEC                                         DECWRITER
TTY2-REP.               R75                                         AV-DEPARTMENT       FAULTY CAR. RETURN


 7  OF THE  7  ITEMS IN THE FILE MATCHED THE REQUIREMENTS
```

ANY TIME YOU ENTER AN UNACCEPTABLE ANSWER, THE QUESTION IS REPEATED.

NOTE THAT THE FIRST LETTER OF A YES/NO RESPONSE IS ACCEPTABLE.

LET'S GET A LISTING OF WHAT IS IN THE FILE

← PRINT EVERYTHING

```
CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?1
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?N
```

← NOTE THAT YOU ARE NOT ASKED IF YOU NEED HELP. IF YOU HAD WANTED HELP, YOU SHOULD HAVE CHOSEN

```
DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?N
```

CHOICE 2) ABOVE, AND SPECIFIED THE SAME FILE.

```
NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?2
SEARCH TYPE # ?2
STANDARD ?R
```

(WE DO NOT REALLY NEED TO SPECIFY THIS CONDITION.)

```
HEADING # ?2
SEARCH TYPE # ?1
STANDARD ?R75

HEADING # ?2
SEARCH TYPE # ?4
STANDARD ?R45

HEADING # ?0
```

WE ARE TRYING TO IDENTIFY ALL OF THE ITEMS THAT NOT ONLY NEEDED TO BE REPAIRED, BUT FOR WHICH THE COST OF THE REPAIRS WAS BETWEEN $45 AND $75.

```
1 FOR ALL CONDITIONS TO BE SATISFIED, 2 IF ANY WILL SUFFICE ?1

NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
?1
?3
?4
?5
?0
```

WE WANT THE ITEM, REPAIR COST, REPAIRER, AND A DESCRIPTION OF WHAT WAS REPAIRED

```
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?N
WHAT IS YOUR NAME ?VIK MUIZNIEKS
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?11/10/76
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?100\0\0
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?
```

USED TO CORRECT A TYPING MISTAKE.

```
FILE--PARTS   LISTED BY VIK MUIZNIEKS ON 11/10/76

ITEM              LINE PRINTER REP. 2
RELATED COSTS-$'S                 R50
REPAIR SERVICE    IBM
DESCRIPTION       BROKEN PRINT HEADS

ITEM              TTY2-REP.
RELATED COSTS-$'S                 R75
REPAIR SERVICE    AV-DEPARTMENT
DESCRIPTION       FAULTY CAR. RETURN
```

THIS IN A FORM WHICH COULD BE STORED IN SEPARATE FILES FOR EACH ITEM (AFTER BEING TORN OFF).

```
  2  OF THE  7  ITEMS IN THE FILE MATCHED THE REQUIREMENTS

CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?1
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?N

DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?N

NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?2
SEARCH TYPE # ?2
HEADING # ?0
```

LET'S GET THE INITIAL COST OF THE ITEMS.

```
NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
?1
?5
?0
```

WE ARE ONLY INTERESTED IN THE ITEMS AND THEIR COSTS

```
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?Y
WHAT IS YOUR NAME ?VIK MUIZNIEKS
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?NOVEMBER 10, 1976
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?80
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?

FILE--PARTS   LISTED BY VIK MUIZNIEKS ON NOVEMBER 10

ITEM                    RELATED COSTS-$'S

CARD READER             3000
LINE PRINTER            5000
TTY1                    1000
TTY2                    1500
```

HERE IS A COMMA, ANYTHING AFTER THE COMMA IS IGNORED (IN THIS CASE 1976).


```
  4  OF THE  7  ITEMS IN THE FILE MATCHED THE REQUIREMENTS
```

```
TOTAL UNDER HEADING RELATED COSTS-0'S =  10500
```

D3

```
CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?2
WHICH FILE DO YOU WANT? (TYPE 'NONE' FOR NONE)
CHOICES:
PARTS           CLASS
 ?CLASS

PASSWORD ?GRADES  <-- THIS FILE HAS A PASSWORD ASSOCIATED WITH IT.
TRY AGAIN, PASSWORD ?SCORES  <-- HERE IT IS.
DO YOU NEED HELP(YES OR NO) ?N  <-- NOTE THAT WE ARE ASKED IF WE NEED HELP, SINCE WE
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?Y                    ASKED FOR A NEW FILE.

HERE ARE THE HEADINGS (AND THEIR CORRESPONDING NUMBERS)
 1              NAME
 2              SEX
 3              TEST1
 4              TEST2
 5              TEST3
 6              GRADE


DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?N
NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?1
SEARCH TYPE # ?3                 } LET'S SEE WHAT IS IN
STANDARD ?ZZZ                    }
                                 } THIS FILE.
HEADING # ?0                     }

NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
 ?-1  <-- PRINT EVERYTHING
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?Y
WHAT IS YOUR NAME ?VIK MUIZNIEKS
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?NOVEMBER 10 1976
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?120
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?

FILE--CLASS   LISTED BY VIK MUIZNIEKS ON NOVEMBER 10 1976

NAME                 SEX                  TEST1                TEST2                TEST3                GRADE

ACE                  MALE                 100                  100                  100                  A
CONSTANCE            FEMALE               81                   82                   82                   B
FLORA                FEMALE               27                   32                   48                   F
HOPE                 FEMALE               62                   67                   59                   D
NORM                 MALE                 75                   71                   78                   C
RANDOM               FEMALE               73                   55                   92                   C


 6  OF THE  6  ITEMS IN THE FILE MATCHED THE REQUIREMENTS


CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?1
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?N

DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?N


NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?2
SEARCH TYPE # ?1                 }
STANDARD ?FEMALE                 }
                                 }
HEADING # ?6                     }
SEARCH TYPE # ?3                 }  WE ARE ASKING FOR ALL OF THE
STANDARD ?C                      }
                                 }  FEMALE STUDENTS WHO HAVE A
HEADING # ?6                     }
SEARCH TYPE # ?4                 }  GRADE OF "B" OR "C".
STANDARD ?B                      }
                                 }
HEADING # ?0                     }

1 FOR ALL CONDITIONS TO BE SATISFIED, 2 IF ANY WILL SUFFICE ?1


NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
 ?6   }
 ?1   } NOTE THAT WE DO NOT HAVE TO SPECIFY THE HEADING #'S IN
 ?0   } ORDER, THEY WILL BE PRINTED IN THE PROPER ORDER AUTOMATICALLY. (IF WE SPECIFY
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?Y          A HEADING # MORE THAN ONCE, IT
WHAT IS YOUR NAME ?VIK MUIZNIEKS                                    IS ONLY PRINTED ONCE).
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?NOVEMBER 10 1976
```

```
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?80
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?
```

04

```
FILE--CLASS   LISTED BY VIK MUIZNIEKS ON NOVEMBER 10 1976

NAME                    GRADE

CONSTANCE               B
RANDOM                  C


 2  OF THE  6  ITEMS IN THE FILE MATCHED THE REQUIREMENTS


CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?1
DO YOU WANT A LIST OF HEADINGS(YES OR NO) ?N

DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO) ?N


NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING
THAT YOU WANT. ENTER 0 FOR HEADING WHEN DONE.

HEADING # ?3
SEARCH TYPE # ?5

HEADING # ?5
SEARCH TYPE # ?5

HEADING # ?4
SEARCH TYPE # ?5

HEADING # ?6
SEARCH TYPE # ?5

HEADING # ?0
```

WE ARE INTERESTED IN SEEING HOW THE CLASS DID A WHOLE ON THE 3 TESTS. HEADING #6 IS ACCIDENTALLY INCLUDED BUT THIS DOES NOT HURT ANYTHING. NOTE THAT THE HEADINGS #'S NEED NOT BE SPECIFIED IN ANY PARTICULAR ORDER.

```
1 FOR ALL CONDITIONS TO BE SATISFIED, 2 IF ANY WILL SUFFICE ?2


NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED
ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS
?1
?3
?5
?4
?0
```

WE ARE NOT INTERESTED IN THE SEX OF THE STUDENTS, OR THEIR GRADES.

```
IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES' ?YES
WHAT IS YOUR NAME ?VIK MUIZNIEKS
WHAT IS TODAY'S DATE(NO COMMAS PLEASE) ?11/10/76
HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT ?100
POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE ?

FILE--CLASS   LISTED BY VIK MUIZNIEKS ON 11/10/76

NAME                    TEST1                   TEST2                   TEST3

ACE                     100                     100                     100
CONSTANCE               81                      82                      82
FLORA                   27                      32                      48
HOPE                    62                      67                      59
NORM                    75                      71                      78
RANDOM                  73                      55                      92


 6  OF THE  6  ITEMS IN THE FILE MATCHED THE REQUIREMENTS


TOTAL UNDER HEADING TEST1 =  418
TOTAL UNDER HEADING TEST3 =  459
TOTAL UNDER HEADING TEST2 =  407
TOTAL UNDER HEADING GRADE =  0
```

HERE IS WHAT WE ARE REALLY LOOKING FOR IN THIS SEARCH

```
CHOOSE ONE OF THE FOLLOWING PLEASE:
   1)  SEARCH FOR SOMETHING ELSE IN SAME FILE
   2)  SEARCH IN A NEW FILE
   3)  RETURN TO OTHER INFO PROGRAM OPTIONS
   4)  LEAVE COMPLETELY

WHAT IS THE NUMBER OF YOUR CHOICE ?4


TIME:  9.26 SECS.

READY
```

APPENDIXE

LISTING OF INFOSH

```
10 REM ALLOWS FOR SEARCHES UNDER DESIGNATED HEADINGS
20 REM WRITTEN BY VIK MUIZNIEKS 10/76
30 REM PRINT STATEMENTS SERVE TO IDENTIFY WHAT IS GOING ON WHEN NO
40 REM COMMENTS EXIST.
50 DIM S$(50),E$(50),H(50),B$(50),R(50)
60 DIM V(50),W(50)
70 R$=" 0123456789"
80 PRINT "WHICH FILE DO YOU WANT? (TYPE 'NONE' FOR NONE)"
90 V7=0
100 MARGIN 132
110 FILE #4,"INFOLS"
120 PRINT "CHOICES:"
130 RESTORE #4
140 IF END #4,THEN 180
150 READ #4,A$
160 PRINT A$,
170 GO TO 140
180 PRINT
190 INPUT F$
200 PRINT
210 IF F$="NONE" THEN 2190
220 RESTORE #4
230 IF END #4,THEN 120
240 READ #4,A$
250 IF A$=F$ THEN 270
260 GO TO 230
270 FILE #3,F$
280 RESTORE #3
290 QUOTE #3
300 READ #3,P$,N$,A$
310 IF P$="" THEN 430
320 PRINT "PASSWORD";
330 INPUT D$
340 IF D$=P$ THEN 430
350 PRINT "TRY AGAIN, PASSWORD";
360 INPUT D$
370 IF D$<>P$THEN 80
375 GOTO 430
380 PRINT "DO YOU NEED HELP(YES OR NO)";
390 INPUT Q$
400IF LEFT$(Q$,1)="Y" THEN 2840
410 IF LEFT$(Q$,1)="N" THEN 470
420 GOTO 380
430 FOR S1=1 TO 50
440 READ #3,S$(S1)
450 IF END#3, THEN 380
460 NEXT S1
470 PRINT "DO YOU WANT A LIST OF HEADINGS(YES OR NO)";
480 INPUT Q$
490 IF LEFT$(Q$,1)="Y" THEN 520
500 IF LEFT$(Q$,1)="N" THEN 590
510 GOTO 470
520 PRINT
530 PRINT "HERE ARE THE HEADINGS (AND THEIR CORRESPONDING NUMBERS)";
540 PRINT
550 FOR J=1 TO S1
560 PRINT J,S$(J)
570 NEXT J
580 PRINT
590 PRINT
600 PRINT "DO YOU NEED A LIST OF SEARCH TYPES(YES OR NO)";
610 INPUT Q$
620 IF LEFT$(Q$,1)="Y" THEN 650
630 IF LEFT$(Q$,1)="N" THEN 690
640 GOTO 600
650 PRINT
660 PRINT
670 PRINT "CHOICES: 1) EXACT MATCH  2) BEGINNING MATCHES STANDARD"
680 PRINT "3)  <= STANDARD  4)  >= STANDARD  5)  LOOK FOR NONNEGATIVE INTEGERS"
690 N4=N5=N6=N7=N8=N9=V4=V7=0
700 PRINT
710 PRINT
720 PRINT "NOW SPECIFY THE HEADINGS AND TYPE OF SEARCH FOR EACH HEADING"
730 PRINT "THAT YOU WANT; ENTER 0 FOR HEADING WHEN DONE."
740 FOR I1=1 TO 50
750 PRINT
760 PRINT "HEADING #";
770 INPUT H(I1)
780 IF H(I1)=0 THEN 960
790 IF H(I1)<1 THEN 930
800 IF H(I1)>S1 THEN 930
810 PRINT "SEARCH TYPE #";
820 INPUT R(I1)
830 IF R(I1)<1 THEN 910
840 IF R(I1)>5 THEN 910
850 IF R(I1)<>5 THEN 880
860 V4=1
870 GOTO 950
880 PRINT "STANDARD";
890 INPUT B$(I1)
900 GOTO 950
910 PRINT "CHOICES ARE 1-5, TRY AGAIN"
920 GOTO 820
930 PRINT "THAT HEADING NUMBER DOESN'T EXIST, TRY AGAIN";
940 GOTO 770
950 NEXT I1
960 C=1
970 I1=I1-1
980 IF I1=1 THEN 1050
990 PRINT
1000 PRINT
1010 PRINT "1 FOR ALL CONDITIONS TO BE SATISFIED, 2 IF ANY WILL SUFFICE";
1020 INPUT C
1030 IF C<1 THEN 1010
1040 IF C>2 THEN 1010
1050 GOSUB 2450
1060 GOSUB 2120
1070 REM NOW BEGIN SEARCHING THROUGH THE FILE
1080 FILE #1,F$+"F"
1090 RESTORE #1
1100 IF END #1, THEN 2190
1110 FOR S2 = 1 TO S1
1120 INPUT #1, E$(S2)
1130 IF E$(S2)="" THEN 1150
1140 N9=1
1150 NEXT S2
1160 IF N9=0 THEN 1190
1170 N8=N8+1
1180 REM DETERMINE THE TYPE OF SEARCH BEING CONDUCTED
1190 FORI = 1 TOI1
1200 ON R(I) GOTO 1220,1250,1280,1310,2740
1210 REM LOOK FOR EXACT MATCH WITH STANDARD
1220 IF B$(I) = E$(H(I)) THEN 1360
1230 GOTO 1320
1240 REM LOOK FOR SAME BEGINNING ON LEFT SIDE OF DATA AS STANDARD
1250 IF B$(I) = LEFT$(E$(H(I)),LEN(B$(I))) THEN 1360
1260  GO TO 1320
1270 REM SEE IF DATA ITEM IS <= STANDARD
1280 IF B$(I)>= E$(H(I)) THEN 1360
1290 GOTO 1320
1300 REM SEE IF DATA ITEM IS >= STANDARD
1310 IF B$(I) <= E$(H(I)) THEN 1360
1315 REM SEE IF SEARCH COMPLETE
1320 IF C=1 THEN 1100
1330 IF I=I1 THEN 1100
1340 GO TO 1520
1360 IF C=1 THEN 1510
1370 REM COMPUTE COLUMN TOTALS FOR ITEMS THAT MEET REQUIREMENTS OF SEARCH
1380 IF I=I1 THEN 1540
1390 FOR J=I+1 TO I1
1400 IF R(J)<>5 THEN 1490
1410 FOR V6=1 TO LEN(E$(H(J)))
1420 FOR V5=1 TO 11
1430 IF MID$(E$(H(J)),V6,1)=MID$(R$,V5,1) THEN 1460
1440 NEXT V5
1450 GOTO 1490
1460 NEXT V6
1470 IF E$(H(J))="" THEN 1490
1480 W(J)=W(J)+VAL(E$(H(J)))
1490 NEXT J
1500 GOTO 1540
1510 IF I=I1 THEN 1540
1520 NEXT I
1530 REM DETERMINE WHAT INFORMATION TO PRINT OUT
1540 IF V7=0 THEN 1650
1550 IF N7=1 THEN 1800
1560 REM PRINT OUT ROW FORM
1570 FOR S2=1 TO S1
```

```
1580 IF V(S2)=0 THEN 1600
1590 PRINT S$(S2),E$(S2)
1600 NEXT S2
1610 N6=N6+1
1620 PRINT
1630 GO TO 1100
1640 REM OBTAIN DEMOGRAPHIC INFORMATION RELEVANT TO PRINT OUT
1650 PRINT "WHAT IS YOUR NAME";
1660 INPUT Y$
1670 PRINT "WHAT IS TODAY'S DATE(NO COMMAS PLEASE)";
1680 INPUT X$
1690 PRINT "HOW MANY CHARACTERS PER LINE(1-132) ARE TO BE PRINTED IN OUTPUT";
1700 INPUT N3
1710 IF N3<1 THEN 1690
1720 IF N3>132 THEN 1690
1730 PRINT "POSITION PAPER TO TOP OF PAGE AND PRESS RETURN PLEASE";
1740 INPUT W$
1750 PRINT
1760 PRINT "FILE--";F$;"   LISTED BY ";Y$;" ON ";X$
1770 PRINT
1780 V7=1
1790 IF N7=0 THEN 1570
1800 IF N4=1 THEN 2010
1810 N4=1
1820 REM SEE IF NEED TO PRINT OUT COLUMN HEADING NAMES
1830 IF N5<>0 THEN 1850
1840 N5=1
1850 N3=INT(N3/N5)
1860 P=VAL(N$)
1870 IF N3<P+2 THEN 1890
1880 LET N3=P+2
1890 N2=0
1900 V2=1
1910 IF V(V2)=0 THEN 1950
1920 IF S$(V2)="" THEN 1940
1930 PRINT TAB(N3*N2);LEFT$(S$(V2),N3);
1940 N2=N2+1
1950 V2=V2+1
1960 IF N2=N5 THEN 1980
1970 GOTO 1910
1980 PRINT
1990 PRINT
2000 REM PRINT OUT DATA ITEMS UNDER PROPER COLUMN HEADINGS
2010 N2=0
2020 FOR S2=1 TO S1
2030 IF V(S2)=0 THEN 2070
2040 IF E$(S2)=""THEN 2060
2050 PRINT TAB(N3*N2);LEFT$(E$(S2),N3-1);
2060 N2=N2+1
2070 NEXT S2
2080 PRINT
2090 N6=N6+1
2100 GOTO 1100
2110 REM PRINT OUT TOTAL INFORMATION FOR SEARCH AND COLUMNS
2120 PRINT "IF YOU WANT THE RESULTS LISTED IN COLUMN FORM TYPE 'YES'";
2130 INPUT Z$
2140 IF Z$="YES" THEN 2170
2150 IF Z$="Y" THEN 2170
2160 GOTO 2180
2170 N7=1
2180 RETURN
2190 PRINT
2200 PRINT
2210 PRINT N6;" OF THE ";N8;" ITEMS IN THE FILE MATCHED THE REQUIREMENTS"
2220 PRINT
2230 PRINT
2240 IF V4=0 THEN 2330
2250 FOR V5=1 TO I1
2260 IF R(V5)<5 THEN 2280
2270 PRINT "TOTAL UNDER HEADING ";S$(H(V5));" = ";W(V5)
2280 NEXT V5
2290 PRINT
2300 PRINT
2310 PRINT
2320 PRINT
2330 PRINT "CHOOSE ONE OF THE FOLLOWING PLEASE:"
2340 PRINT "    1)  SEARCH FOR SOMETHING ELSE IN SAME FILE"
2350 PRINT "    2)  SEARCH IN A NEW FILE"
2360 PRINT "    3)  RETURN TO OTHER INFO PROGRAM OPTIONS"
```



```
2380 PRINT "    4)  LEAVE COMPLETELY"
2390 PRINT
2400 PRINT "WHAT IS THE NUMBER OF YOUR CHOICE";
2410 INPUT N9
2420 IF N9<1 GOTO 2190
2430 IF N9>4 THEN 2190
2440 ON N9 GOTO 470,80,2710,3180
2450 FOR V1=1 TO 50
2460 V(V1)=0
2470 W(V1)=0
2480 NEXT V1
2490 PRINT
2500 PRINT
2510 PRINT "NOW SPECIFY THE NUMBERS OF ALL THE HEADINGS YOU WANT PRINTED"
2520 PRINT "ENTER 0 WHEN DONE OR -1 FOR ALL THE HEADINGS"
2530 FOR V1=1 TO 50
2540 INPUT N
2550 IF N=-1 THEN 2660
2560 IF N=0 THEN 2700
2570 IF N<1 THEN 2630
2580 IF N>S1 THEN 2630
2590 IF V(N)=1 THEN 2640
2600 V(N)=1
2610 N5=N5+1
2620 GOTO 2640
2630 PRINT "ENTER ONLY -1, 0  OR A NUMBER CORRESPONDING TO A HEADING"
2640 NEXT V1
2650 GOTO 2700
2660 FOR V1=1 TO 50
2670 V(V1)=1
2680 NEXT V1
2690 N5=S1
2700 RETURN
2710 CHAIN "INFO"
2730 REM SEE IF DATA ITEM IS NONNEGATIVE INTEGER, AND UPDATE COL. TOTAL
2740 FOR V6=1 TO LEN(E$(H(I)))
2750 FOR V5=1 TO 11
2760 IF MID$(E$(H(I)),V6,1)=MID$(R$,V5,1) THEN 2790
2770 NEXT V5
2780 GOTO 1320
2790 NEXT V6
2800 IF E$(H(I))="" THEN 1320
2810 W(I)=W(I)+VAL(E$(H(I)))
2820 GOTO 1360
2830 REM HELP SEQUENCE
2840 PRINT
2850 PRINT "YOU WILL BE ASKED TO SPECIFY THE HEADING #'S OF ALL THE HEADINGS"
2860 PRINT "YOU WANT TO SEARCH UNDER, AND THE PARTICULAR TYPE OF SEARCH YOU"
2870 PRINT "WANT TO CONDUCT.  THE SEARCHES CAN BE OF THE TYPE (1) LOOK FOR ALL"
2880 PRINT "EXACT MATCHES TO A SPECIFIED STANDARD UNDER THAT HEADING"
2890 PRINT "(2) LOOK FOR ALL ITEMS UNDER THE HEADING THAT HAVE THE SAME FIRST"
2900 PRINT "LETTERS AS A SPECIFIED STANDARD  (3) LOOK FOR ALL ITEMS UNDER THE"
2910 PRINT "HEADING THAT HAVE A VALUE LESS THAN OR EQUAL TO A SPECIFIED STANDARD"
2920 PRINT "(4) LOOK FOR ALL ITEMS UNDER THE HEADING THAT HAVE A VALUE GREATER"
2930 PRINT "THAN OR EQUAL TO A SPECIFIED STANDARD  (5) LOOK FOR ALL ITEMS UNDER"
2940 PRINT "THE HEADING THAT HAVE A NUMERICAL EQUIVALENT VALUE; BE CAREFUL"
2950 PRINT "IF AN ENTRY CONTAINS 2 NUMBERS WITH BLANKS INBETWEEN, THEY WILL BE"
2960 PRINT "TREATED AS ONE NUMBER--E.G.  '12  34' WOULD BE INTERPRETED AS 1234."
2970 PRINT "AS A SIDE BONUS, YOU GET THE TOTAL FOR ALL OF THE NUMBERS FOUND"
2980 PRINT "UNDER THIS HEADING THAT MEET THE REST OF THE SEARCH CRITERIA (IF ANY)."
2990 PRINT "  ONCE YOU HAVE SPECIFIED ALL OF THE HEADINGS, SEARCH TYPES, AND"
3000 PRINT "RELATED STANDARDS, YOU MUST ALSO INDICATE (1) IF ALL OF THE"
3010 PRINT "CONDITIONS SPECIFIED MUST BE MET BEFORE ANY ITEM IS INCLUDED, OR IF"
3020 PRINT "(2) MEETING ANY OF THE CRITERIA IS ENOUGH TO INCLUDE AN ITEM"
3030 PRINT "(UNLESS YOU ARE ONLY SEARCHING UNDER ONE HEADING).  ALSO, YOU ARE"
3040 PRINT "GIVEN THE CHOICE AS TO WHICH HEADINGS TO HAVE DISPLAYED FOR THE"
3050 PRINT "ITEMS THAT MATCH THE ABOVE SPECIFICATIONS (THEY NEED NOT BE THE SAME"
3060 PRINT "AS THE HEADINGS SEARCHED UNDER).  THE PROGRAM WILL ALSO ASK YOU IF"
3070 PRINT "YOU WANT THE RESULTS LISTED IN TABULAR FORM (AS OPPOSED TO LINE FORM)"
3080 PRINT "AND THE NUMBER OF CHARACTERS PER LINE THAT CAN BE PRINTED (1-132)"
3090 PRINT "AS WELL AS YOUR NAME AND THE DATE.  IT ALLOWS YOU TO POSITION THE"
3100 PRINT "PAPER TO THE TOP OF A NEW PAGE OF PAPER BEFORE PRINTING THE RESULTS OF"
3110 PRINT "THE SEARCH, AND THEN GIVES YOU THE OPTIONS OF (1) SEARCHING THE SAME"
3120 PRINT "FILE AGAIN  (2) SEARCHING ANOTHER FILE  (3) RETURNING TO THE"
3130 PRINT "OTHER INFO PROGRAM OPTIONS  (4)  LEAVE THE PROGRAM AND RETURN TO"
3140 PRINT "THE BASIC MONITOR."
3150 PRINT
3160 PRINT
3170 GOTO 470
3180 END
```

ER